CEWES MSRC/PET TR/99-34

CBay VisGen User Guide

Version 1.0

by

Rob Stein

07h00299

Work funded by the DoD High Performance Computing Modernization Program CEWES Major Shared Resource Center through

Programming Environment and Training (PET)

Supported by Contract Number: DAHC94-96-C0002
Nichols Research Corporation

CBayVisGen User Guide

Version 1.0

Rob Stein
NCSA
University of Illinois

March 1999

Contents

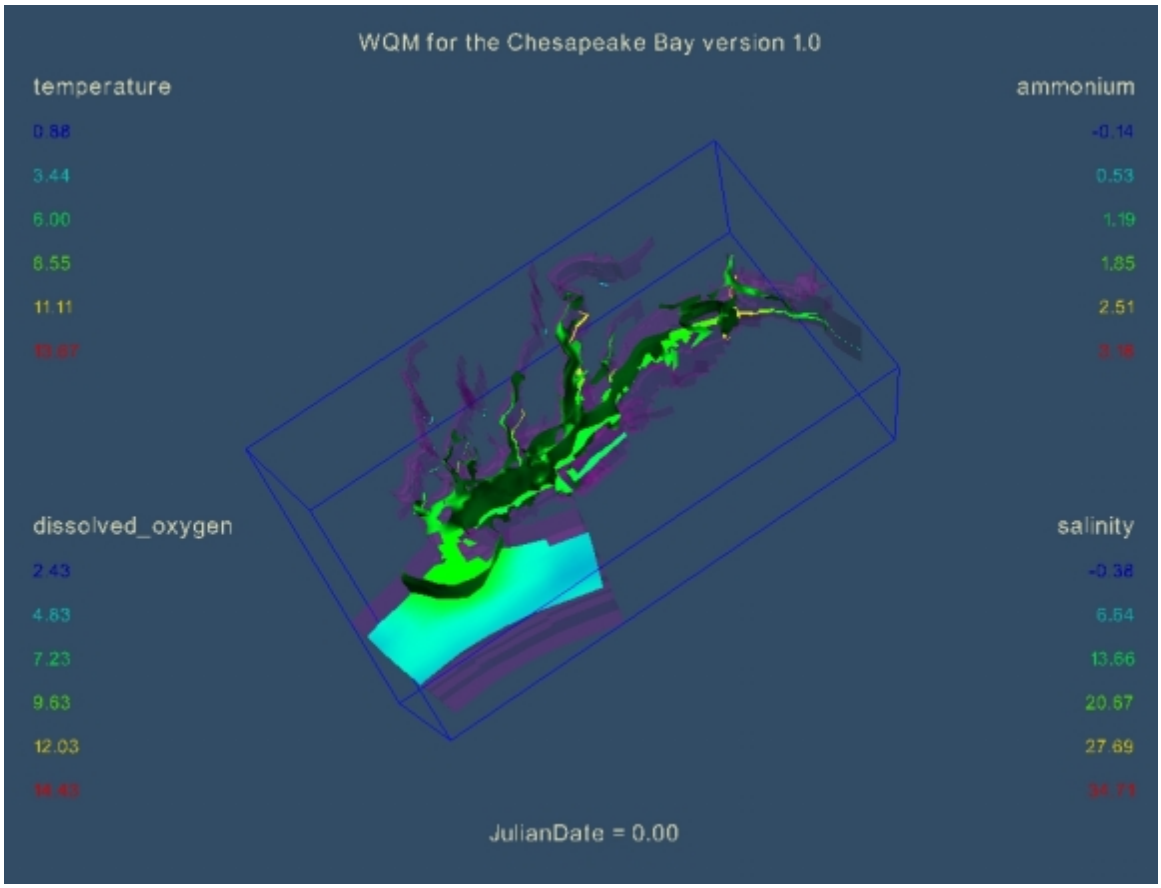

Figure 1: The CBayVisGen Display Window.

1 Introduction

The Chesapeake Bay is the largest and most productive estuary in the United States. Population growth and the development of agriculture and industry in the area surrounding the Bay have had significant impact. CBayVisGen is a visualization tool designed to support display and exploration of simulation output produced by the CE-QUAL-ICM code. This user guide outlines how to use CBayVisGen.

2 Display Window

The display window for CBayVisGen has several major sections, as shown in Fig. 1. This window contains the graphical representations of the different visualizations, or VizNodes, that you have added into the scene. It may also contain a blue box denoting the bounding area of the data set for the bay. This outline can be added or removed in the Config File (see Section 5). The view can be rotated and manipulated with the NavPad (see Graphical User

Interface description, below). Two other sections of this window are the Title Text and the Time Text sections. These add some annotation capabilities to the window. The contents and size of both of these sections is controlled though the configuration file. The last of the main sections of this window are the various color scales that appear on the sides of the window. As a default, no color scales appear. Scales can be added to the main window through the DataNode Widget. The scales will automatically position themselves in the window starting in the upper left-hand corner and proceeding counter-clockwise. You can use up to 4 color scales at a time. The size of the color scales can also be determined in the configuration file.

3 Graphical User Interface

When the application starts up, the main interface window appears, shown in Fig. 2. This window is divided into four sections. The first two sections handle the creation, deletion, and modification of VizNodes and DataNodes. The user can also control the time step of the data set that they are viewing from the Time Control section. The orientation and position of the bay can be controlled with the NavPad from the Model Control section. Lastly the user also has several options for capturing the current graphics state either as an image, a movie, or one of several other representations.

3.1 VizNode Control

A VizNode is an object in the application that uses data from a DataNode, and translates the data into a particular visual representation. For example: slices through the data set, isosufaces of scalar values, and an external volume representation are all VizNodes.

Adding a VizNode is easy. First click on the *Add* button in the VizNode section of the Main GUI window. This will bring up a dialog box, shown in Fig. 3.

In this dialog box you will need to select which type of VizNode you would like to create from the list of available types. You will then need to select which DataNode to operate on. The DataNode list shows all the DataNodes currently in the system. Therefore if there are no DataNodes in the system, there will be no data to create a VizNode from. After selecting the data that you want to work with, give the new VizNode a name in the name entry box. A name is automatically suggested, but you can change this to suit your needs. Remember you need to ensure that each VizNode has a unique name.

3.2 DataNode Control

A DataNode is an object in the application that is responsible for holding and managing individual data sets. (e.g., Salinity, Temperature, Dissolved_Oxygen). The Data Node control is shown in Fig. 4. Among other things, the DataNode manages the range of data

Figure 2: The main interface window contains (from top to bottom) the Viz Node Control, the Data Node Control, the Time Animation Control, the Model Manipulation Control, and the Capture Control.

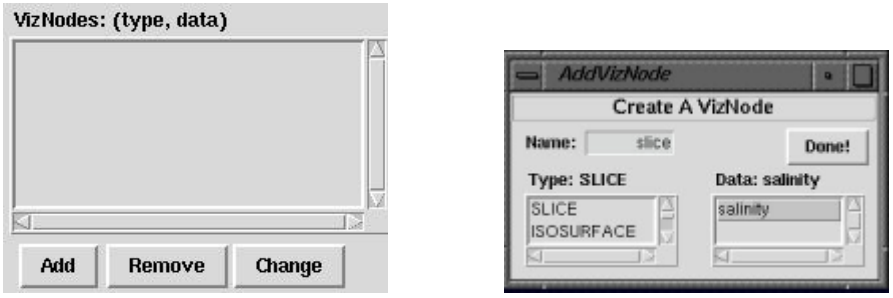

Figure 3: On the left, the VizNode section from the main graphical user interface. On the right, the dialog box to add a new VizNode.

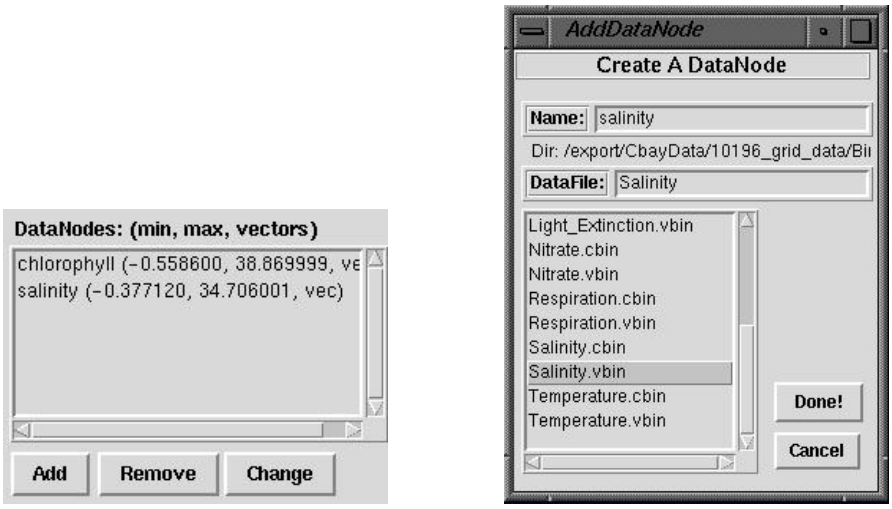

Figure 4: On the left, the data node section from the main graphical user interface. On the right, the dialog box to add a new data node.

that is mapped to color in the application. This ensures that two VizNodes using the same DataNode will represent the color of that data in the same way.

Bringing data into the session is easy. When you click on the *Add* button of the DataNode section of the main user interface, you will see the dialog box shown in Fig. 4. Using this dialog box, use the file list and navigate to where you are storing the binary data files for this application (for example, "./Data/Binaries"). For more information on binary data files, see the section on Creating Data Files (Sec. 7). In this directory, you should see files with extensions that are either *.cbin or *.vbin. These files store the data in either cell-centered, or vertex-centered format respectively. You can select either of these files types. The name of the data file that you selected will appear in the Data File entry box. Note that the extension is not present. You can give this DataNode whatever name you would like in the name entry box. An initial name is suggested. After doing these things, click the *Done* button, and after a little while you should see the DataNode you created appear in the DataNode section of the main user interface.

3.3 Time Animation Control

The data from the simulation contains multiple time steps, or Julian Days. CBayVisGen allows you to control which Julian Day you are viewing and it allows you to animate through some sequence of Julian Days. In Fig. 5, you see the Time Animation section of the main user interface window. Notice the slider in the middle of this section. This slider allows for interactive control of the time step. Dragging this slider back and forth will change the time step. The number on top of the slider represents the current time step. Note: this is not necessarily the same as the Julian Date. The Julian Date is reported in the Display Window.

To automate the playback of the Julian Days, you can use the Time Animation facilities provided. At the top of the Time Animation section, there are three entry boxes, which control the parameters for playback. The first box sets the starting time step, the second box sets the ending time step and the last box sets the step or stride that you want to take between the start and end. This controls the number of frames in the animation. Once these entries are set, just press the play button. The application will then animate through all the time steps specified.

3.4 Model Manipulation Control

To control your view of the Chesapeake Bay, you can use the *NavPad* provided in the Model Manipulation section of the user interface and shown in Fig. 6. The NavPad is very simple to use. Move your mouse inside the red circle on the NavPad, click the first button on your mouse, and drag the circle around. You will notice that your movements on the NavPad are reflected in the Display Window. When you release the mouse button, the red circle returns to the center of the pad. The NavPad is similar to a trackball, in that your movements are accumulated. That is, if you make several movements from the left to the right on the NavPad, the bay model will rotate in a right-handed manner around its y-axis. There are two different navigation modes available for positioning the model. The default is a rotation mode, and the second mode is a translation mode. The translation mode will move the model around in a plane normal to your current viewing direction. You can access the different modes by pressing the rightmost button immediately under the NavPad. The text of this button will reflect your current navigation mode, either *Rotate* or *Translate*. You can also zoom in a section of the model. The plus and minus buttons under the NavPad will cause your view to zoom in and out by a pre-defined amount. To return to the default view, just press the *Reset* button.

3.5 Capture Control

Some of the most useful features of the application are the abilities to capture your graphics in different formats. In particular, you can capture the graphics as an image or a movie file.

You can capture a snapshot of the Display Window at any time. All you need to do is click on the *Snapshot* button in the Model Manipulation section of the main user interface. This will bring up the Export dialog box, shown in Fig. 7.

To export the scene, use the File Selection box to go to the directory where you want to store the captured file. Type the filename that you want to use into the Name entry box. Then select the type of file you would like to export. A JPEG image is currently the default, but you can also export the scene to a VRML format as well as a RIB format. The RIB format can be fed to rendering systems that understand the RenderMan API – most of these systems provide high-quality rendering. Finally, click the *Done* button.

The second option for capturing graphics in the application is to capture a movie of

what's going on. To do this, simply click the *Movie* button in the Model Manipulation section of the main user interface window. The dialog box shown in Fig. 7 will appear.

To start capturing a movie, use the File Selection box to navigate to the directory where you want to store your captured movie file. Next, type in the name of the movie file into the Name entry box. The movie that will be exported will be in QuickTime format, so a *.mov extension is recommended for the filename. Next, click on the green *Start* button to start recording the movie. From this moment on the graphics in the Display Window will be captured into a movie file. The movie is being created while you work, so your frame rate will be slower. One common thing to do would be to start a Time Animation, and record this as a movie. The application automatically stops any movie from recording when the time series has been exhausted. So if you are recording a long time series, you don't need to sit around and baby-sit the movie-making – the movie will terminate when the time series ends. If you want to end the movie manually, just click the *Stop* button on the dialog box.

4 Using the Tools

The various visual representations provided by CBayVisGen each have their own dialog box for setting parameters. A few of these appear in Fig. 8.

4.1 Slices

The Slice tool cuts the dataset along a plane described by an origin and normal. The data is interpolated to the plane at each point for color-mapping. The user interface controls are as follows. At the top of the dialog box is a label with the name of the VizNode that you are currently affecting (i.e. Configure Node node-name). Just below this, there is a series of radio buttons that allows you to choose one of the Cartesian coordinates as the normal of your cutting plane. There is a slider below the radio buttons. These allow you to choose the position of the plane along the slicing direction. The SliceNodes menu button allows you to toggle through any other slices that you may have currently active. Next to the SliceNodes menu is a toggle button controlling the visibility of the slice plane. The bottommost control in this dialog box is the *TileMode* radio button. This allows you to select whether you would like to see a slice generated from cell-centered data (TileMode is **On**) or one generated from vertex-centered data (TileMode is **Off**). The cell-centered view will result in the slice looking tiled since each cell will have only one color and not a gradient of colors as with the vertex-centered data.

4.2 Isosurfaces

The isosurfacing tool will allow you to generate an isosurface at a particular value in the data set. An isosurface generates geometry from the dataset such that every point on the

Figure 5: The Time control.

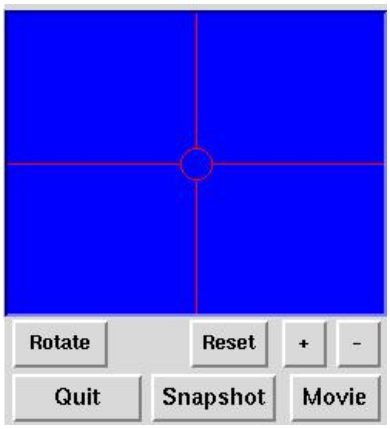

Figure 6: The Navigation control.

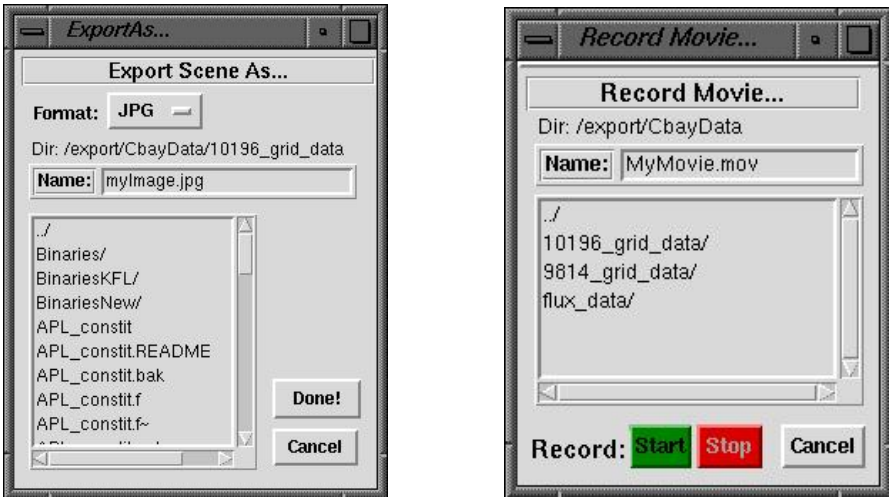

Figure 7: CBayVisGen allows the user to capture a snapshots and movies of the current view.

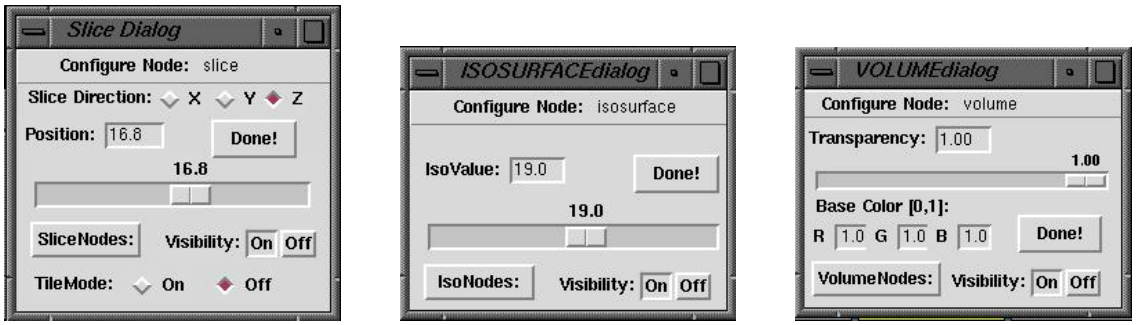

Figure 8: The user interface controls for particular representations each contain a label at the top, widgets specific to choosing parameters for each visualization, and a visibility toggle.

surface is at a constant value. The isosurfacing tool is very easy to use. The Label at the top of the widget tells you which isosurface node you are changing. The *IsoValue* slider is just below this label and can be used to select the value used to generate the surface. Below the slider is the *IsoNodes* menu button. This button allows you to select among existing isosurface nodes. Next to this is a toggle button for controlling visibility of the isosurface. Similar to the slices, this button controls whether or not the isosurface will get computed and drawn. This can be a handy feature, especially for complex scenes.

4.3 Volume Rendering

The volume rendering will extract only the external faces from the dataset and will color code them according to the DataNode that is specified. The volume view can give a good overview of the outer picture of the data. When you make the volume visualization semi-transparent, it provides a good context for the other visual representations.

The interface to the volume node is similar to the two previous nodes. The label appears at the top of the window. Just below this is a slider that allows you to set the transparency value of the volume. Below the transparency slider is a set of entry boxes that determine what color to use in drawing the volume. Like the other two nodes, the listing box labeled *VolumeNodes* allows you to choose among your existing volume nodes, and the *Visibility* toggle button allows you to temporarily turn off the rendering of the volume node. This is particularly handy for this node, since it typically takes a long time to compute and draw. You might want to turn visibility off when adjusting the view of the scene, and reactivate the volume view once the angle of view is set.

4.4 The DataNode Dialog

The DataNode dialog box gives you access to control how your data is displayed and interpreted on the Display Window. In particular, this is where you can select the range of data that is of interest to you. The other thing that you can do from this dialog box is add color

Figure 9: The dialog box that shows information about the data set.

legends to the Display Window. A color legend consists of the name of the data followed by a series of numbers within the range of the data. These numbers are colored according to the colormap that you are currently using. This color scale is also updated when you hit the *Apply* button.

5 Using the Config File

The Chesapeake Bay Application manages many of its startup options through use of a simple configuration file. This file is read in by the application at start up.

The configuration file is a ASCII text file in the same directory as the executable and is named **.cbayrc**. This file can be created and/or edited using your favorite text editor, such as *vi* or *emacs*.

The format of the configuration file is very simple. Each line is a command variable beginning with the variable name (all caps), followed by any amount of white space and the argument for that variable. For example:

```
<MY COMMAND NAME>   My_Variable
```

Here is a listing of the variables that can be set in the configuration file:

GUI This variable controls which Graphical User Interface (GUI) you want to control the application. Its argument is the path and filename of a tcl/tk executable to use as the controlling application

BACKGROUND_COLOR This controls the color of the background of the Main Graphics Window. The arguments are three (3) floating point numbers ranging from 0.0-1.0 in r-g-b order. For example if you wanted the background color to be a gray color your arguments would be ".2 .3 .4"

HUD_TITLE_TEXT This is the variable that controls what text should be displayed at the top of the Main Graphics Window. The argument is a string including all spaces and characters. Note: quotes are not needed.

HUD_TIME_TEXT This is the variable that controls the text that appears next to the Julian Date at the bottom of the Main Graphics Window. The argument is a string including all spaces and characters. Note: quotes are not needed

HUD_TITLE_SIZE This is a variable that describes the relative size of the title text. This number only relates to scale and not to any type of point size like in a word processor. The argument is an integer number. A good suggested value is "12"

HUD_SCALE_SIZE This is a variable that describes the relative size of the Color Scales. This number only relates to scale and not to any type of point size like in a word processor. The argument is an integer number. A good suggested value is "12"

HUD_TIME_SIZE This is a variable that describes the relative size of the time text. This number only relates to scale and not to any type of point size like in a word processor. The argument is an integer number. A good suggested value is "12"

OUTLINE This controls whether or not a blue outline box is displayed representing the bounds of the dataset. This box can help by adding context, but can be visually unappealing for recording movies for example. The argument to this variable is simple 1 or 0 for on and off respectively

MOVIE_FRAME_RATE This variable specifies a rate that you would like to capture your movies at. The argument is an integer number. The higher this number the faster your movie will play, up to the limits of your system. A recommended number is 10.

VERTEX_FILE This variable is used in the creation of new grids only. It defines the file to use to find the latitude and longitude of individual vertices of the bay. This file is normally called node_lat_lon_*.dat

BRICK_FILE This variable is used in the creation of new grids only. It defines the file to use to find the definitions of each cell in the bay. This file is normally called boxnode_*.dat

NUM_TIMESTEPS This variable is used in the creation of new binary data files only. It tells the datafile creation utility how many timesteps it can expect. Its argument is an integer.

NUM_VERTICES This variable is used in the creation of new binary data files only. It tells the datafile creation utility how many vertices it can expect. Its argument is an integer.

NUM_BRICKS This variable is used in the creation of new binary data files only. It tells the datafile creation utility how many cell it can expect. Its argument is an integer.

5.1 Sample Config file

Below is a sample configuration file. It shows all the uses of all the different variables.

```
#CHESAPEAKE BAY APPLICATION CONFIGURATION FILE

#-----------------------------------------------------------------------
#The GUI variable is optional and can additionally be specified on the
#command #line via the -gui option

GUI                     CbayGUI.tcl

#-----------------------------------------------------------------------
#The Background color for the window can be chosen using 3 floating point
#numbers ranging from 0-1 representing red, green, and blue

BACKGROUND_COLOR        .6 .6 .6

#-----------------------------------------------------------------------
#The following are settings that control the appearance and content of
#the text legends that appear around the main viewing area

HUD_TITLE_TEXT WQM for the Chesapeake Bay version 1.0
HUD_TIME_TEXT Julian Date:
HUD_TITLE_SIZE          12
HUD_SCALE_SIZE          12
HUD_TIME_SIZE           1

#-----------------------------------------------------------------------
#Specify if an outline for the dataset should be displayed

OUTLINE 1

#-----------------------------------------------------------------------
#Specify the Frame Rate for capturing movies

MOVIE_FRAME_RATE        10

#-----------------------------------------------------------------------
#Optionally specify which printer to send screen dumps to
PRINTER                 blw2
```

```
#-----------------------------------------------------------------------
#THE FOLLOWING PARAMETERS ARE USED IN THE CREATION OF THE BINARY GRIDS
#AND DATASETS FOR THE APPLICATION

#-----------------------------------------------------------------------

#These are options that you must specify for creating new grids to use
#with the application.  If you are not creating new grids these options
#can be omitted

VERTEX_FILE     Data/node_lat_long_10196_rect.dat
BRICK_FILE      Data/boxnode_10196_rect.dat

#These are options that you must specify for creating new binary datasets...
#If you are not creating new dataset these options can be omitted

NUM_TIMESTEPS           10
NUM_VERTICES            16963
NUM_BRICKS              10196
```

6 Creating Grid Files

The Chesapeake Bay application uses a particular file format to record the location and topology of cells in the bay. Generally there are two files: one for the physical grid model and one for the computational grid model. These files are contained in the Data directory, and are called cbay_phys_grid.vtk and cbay_comp_grid.vtk respectively.

The tool required to create these grids is called **cbay2vtk** and can be found in the Utilities/ directory. This tool has several requirements that must be met in order for a correct grid to be generated.

6.1 Requirements

- You must correctly set the following variables in the .cbayrc file in the root directory of the Cbay application:
  - VERTEX_FILE
  - BRICK_FILE
  - NUM_VERTICES
  - NUM_BRICKS

– NUM_TIMESTEPS

If you are unsure of how to do this, please see the section on "Using the Config File" (Sec. 5) for details.

- The cbay2vtk tool must be run from the root directory of the Cbay application. From the root directory, type **Utilities/cbay2vtk -comp**.
- You can view the results of the grid creation by using the **gridView** executable in the utilities directory. Type **Utilities/gridView -comp**.

7 Creating Data Files

The Chesapeake Bay application uses a special binary format for the different scalar and vector data files. This format stores information about the dimensions of the dataset as well as the number of Julian Days, and the actual dataset. The format was chosen to improve performance over the original wqm output files.

We have provided a tool to use to easily create new data files for use in the Chesapeake Bay Application. The tool is called **CreateBin**, and can be found in the Utilities directory. There is a graphical user interface that can be accessed by typing **Utilities/CreateBin.tcl**. The interface is shown in Fig. 10.

In the upper left-hand corner there is a box listing all the different constituents that are available to extract from the wqm model file. To choose a constituent for conversion, just select it in the box on the left and then press the *Add* key. You should then see this name added to the extraction list. Names can be removed from the extraction list by selecting the name on the list and then pressing the *Remove* button.

When you have finished adding constituents to the extraction list, press the *Start* button to start the process. You should see lots of text scrolling by in the window labeled Output. If at any time you wish to stop the extraction process, press the *Quit/Stop* button.

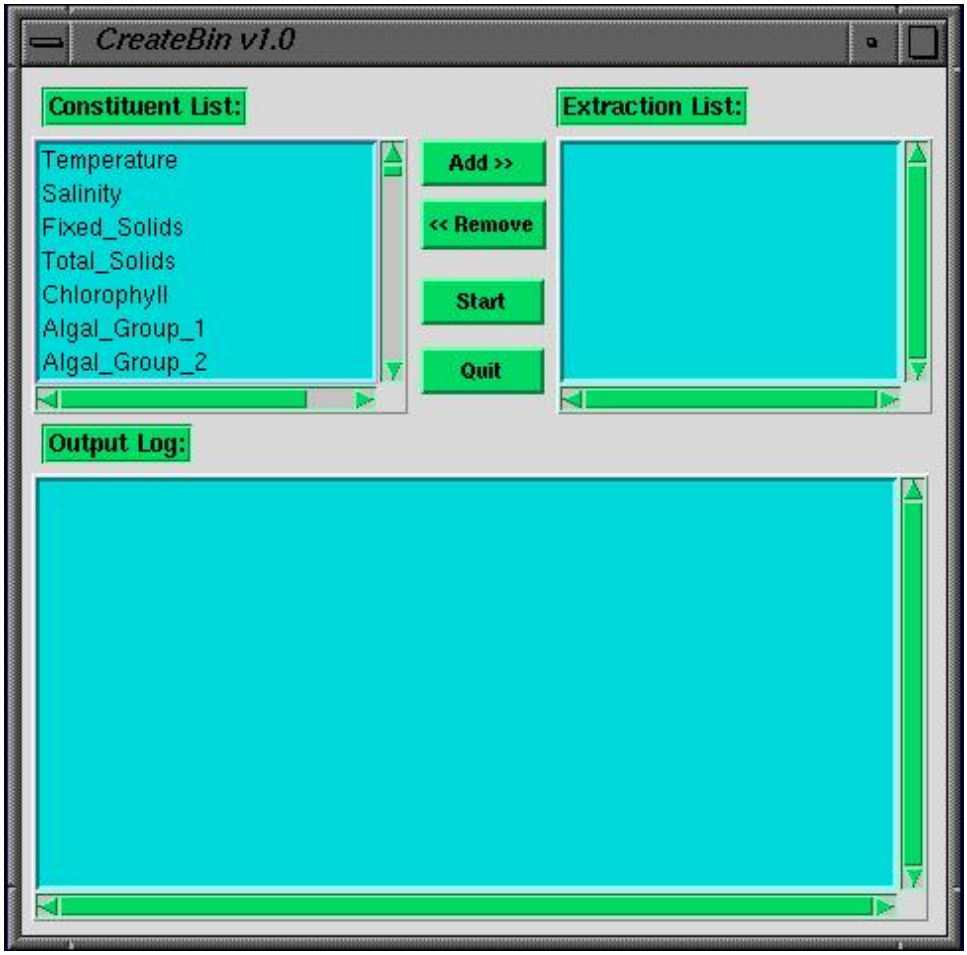

Figure 10: The user interface for creating the binary data.